UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/608,135	06/30/2003	Sanyay Ghemawar	0026-0032	3014

44989 7590 07/21/2006

HARRITY SNYDER, LLP
11350 Random Hills Road
SUITE 600
FAIRFAX, VA 22030

EXAMINER	
THAI, HANH B	

ART UNIT	PAPER NUMBER
2163	

DATE MAILED: 07/21/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/608,135	GHEMAWAR ET AL.	
	Examiner	Art Unit	
	Hanh B. Thai	2163	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *amendment filed 5/2/06*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-17* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-17* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. The following is a Final Office Action in response to the amendment filed May 2, 2006. Independent claims 1, 13, 14, 15, 16 and 17 have been amended. Claims 1-17 are pending in this application.

Response to Arguments

2. Applicant's arguments regarding "serializing performance of the operations" of claims 1-17 have been considered but are moot in view of the new ground(s) of rejection.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claim 16 is rejected under 35 U.S.C. 103(a) as being unpatentable over Adya et al. (US Pub. 2005/0102268 A1) in view of Nitta et al. (US 5,287,521).

Regarding claim 16, Adya discloses a method for concurrently performing first and second operations within a same directory, comprising:

- obtaining a first lock on a sub-directory or file name within the directory by the first operation (¶[0081] and [0118]-[0120]);
- obtaining a second lock on a sub-directory or file name within the directory by the second operation (¶[0118]-[0120]);
- determining whether the first and second locks conflict (¶[0121]-[0123]); and

- concurrently performing the first and second operations when the first and second locks do not conflict, the first and second locks being read-write locks (¶[0123]; [0132]-[0134] and [0137]).

Adya does not disclose serializing performance of the first and second operations when the first lock or the second lock conflicts with the third locks. Nitta discloses method and apparatus for releasing and obtaining shared and exclusive locks including serializing performance of the processes (col.3, lines 41-57 and col.4, line 36 to col.5, line9, Nitta). It would have been obvious to one of ordinary skill in the art at the time of the invention was made to modify Adya to include the claimed feature as taught by Nitta. The motivation of doing so would have been to obtain the lock and execute the processing in the lump can offer higher degree of freedom for processing and easily suppress the overhead (col.3, lines 35-38, Nitta).

4. Claims 1-15 and 17 are rejected under 35 U.S.C. 103(a) as being unpatentable over Adya et al. (US Pub. 2005/0102268 A1) in view of McClaughry et al. (US 5,933,825) and further in view of Nitta et al. (US 5,287,521).

Regarding claim 1, Adya discloses a method for performing operations within a file system in which directories and files are organized as nodes in a namespace tree, the method comprising:

- associating a read-write lock with each of the nodes in the namespace tree (¶ [0033]; [0035]; [0037] and col.12, table II, Adya);
- acquiring a first lock on a name of one or more directories involved in the first operation (¶[0117]-[0119], Adya);

- determining whether the first lock or the second lock conflicts with third locks acquired by a second operation (¶[0121]-[0123], Adya); and
- performing the first operation when the first lock or the second lock does not conflict with the third locks, where the first, second, and third locks are read-write locks (¶[0123]; [0131]-[0133] and [0137], Adya).

Adya, however, does not disclose acquiring a lock on an entire pathname. McClaughry, on the other hand, discloses arbitrating concurrent access to file system objects including locks in the file system object hierarchy (abstract; summary and col.5, line 59 to col.6, line 62, McClaughry) and thus reads on the claimed "lock on an entire pathname". Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention was made to utilize the lock on the entire pathname into the distributed file system of Adya to derive the invention as claimed. The motivation of doing so would have been desirable to provide techniques and systems for managing file system object in a multithreaded environment in which multiple threads can simultaneously operate on the same object whenever possible to maximize throughput and availability of objects to a user but without causing object incoherency and inconsistent file operation results (col.2, lines 50-56, McClaughry).

Adya and McClaughry combination does not disclose serializing performance of the first and second operations when the first lock or the second lock conflicts with the third locks. Nitta discloses method and apparatus for releasing and obtaining shared and exclusive locks including serializing performance of the processes (col.3, lines 41-57 and col.4, line 36 to col.5, line9, Nitta). It would have been obvious to one of ordinary skill in the art at the time

of the invention was made to modify the combination system of Adya and McClaughry to include the claimed feature as taught by Nitta. The motivation of doing so would have been to obtain the lock and execute the processing in the lump can offer higher degree of freedom for processing and easily suppress the overhead (col.3, lines 35-38, Nitta).

Regarding claim 2, Adya/McClaughry combination discloses wherein the performing the first operation includes: concurrently performing the first operation and the second operation when neither the first lock nor the second lock conflicts with the third locks (¶[0123]; [0131]-[0133] and [0137], Adya).

Regarding claim 3, Adya/McClaughry combination discloses wherein the first lock is a read lock (¶[0035]; [0118]-[0119] and [0121]-[0123], Adya).

Regarding claim 4, Adya/McClaughry combination discloses wherein the second lock is one of a read lock and a write lock (¶[0035] and [0121]-[0123], Adya).

Regarding claim 5, Adya/McClaughry combination discloses wherein the first operation is a read operation, the first lock is a read lock, and the second lock is a read lock (¶[0121]-[0123] and 133, Adya).

Regarding claim 6, Adya/McClaughry combination discloses wherein the first operation is a namespace modification operation, the first lock is a read lock, and the second lock is a write lock (¶[0081]-[0085]; and [0121]-[0123], Adya).

Regarding claim 7, Adya/McClaughry combination discloses wherein the first operation is a snapshot operation, the first lock is a read lock, and the second lock is a write lock (summary and col.4, lines 21-32, McClaughry discloses "copy operation" corresponding to "snapshot operation").

Regarding claim 8, Adya/McClaughry combination discloses wherein the determining whether the first lock or the second lock conflicts with third locks includes: using a lazily allocated data structure that maps pathnames to locks to determine whether the first lock or the second lock conflicts with the third locks (¶[0046]; [0102]-[0106]; [0115]; [0123] and [0131]-[0133], Adya).

Regarding claim 9, Adya/McClaughry combination discloses serializing the first, second, and third locks when the first lock or the second lock conflicts with the third locks (¶[0123]; [0131]-[0133] and [0137], Adya).

Regarding claim 10, Adya/McClaughry combination discloses wherein the serializing the first, second, and third locks includes: determining an order for the first, second, and third locks based on levels of the namespace tree involved in the first, second, and third locks and lexicographically within one of the levels of the namespace tree involved in the first, second, and third locks (¶[0115]-[0118], Adya).

Regarding claim 11, Adya/McClaughry combination does not disclose determining an order for the first, second and third locks and permitting the first and second operations to be performed based on the determined order for the first, second and third locks (col.3, lines 41-57 and col.4, line 36 to col.5, line9, Nitta).

Regarding claim 12, Adya/McClaughry combination discloses permitting the first and second operations to concurrently operate within a same one of the directories when neither the first lock nor the second lock conflicts with the third locks (¶[0115]-[0118]; [0123]; [0131]-[0133] and [0137], Adya).

Regarding claim 13, Adya discloses a system for performing operations within a file system, comprising:

- means for obtaining one or more first locks on one or more directory names involved in an operation (¶[0117]-[0119], Adya);
- means for detecting whether the one or more first locks or the second lock conflict with one or more third locks acquired by another operation (¶[0121]-[0123], Adya); and
- means for executing the operation when the one or more first locks or the second lock do not conflict with the one or more third locks, the one or more first locks, the second lock, and the one or more third locks being read-write locks (¶[0123]; [0131]-[0133] and [0137], Adya).

Adya, however, does not disclose means for obtaining a lock on an entire pathname. McClaughry, on the other hand, discloses arbitrating concurrent access to file system objects including locks in the file system object hierarchy (abstract; summary and col.5, line 59 to col.6, line 62, McClaughry) and thus reads on the claimed "lock on an entire pathname". Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention was made to utilize the lock on the entire pathname into the distributed file system of Adya to derive the invention as claimed. The motivation of doing so would have been desirable to provide techniques and systems for managing file system object in a multithreaded environment in which multiple threads can simultaneously operate on the same object whenever possible to maximize throughput and availability of objects to a user but without causing object incoherency and inconsistent file operation results (col.2, lines 50-56, McClaughry).

Adya and McClaughry combination does not disclose serializing performance of the first and second operations when the first lock or the second lock conflicts with the third locks. Nitta discloses method and apparatus for releasing and obtaining shared and exclusive locks including serializing performance of the processes (col.3, lines 41-57 and col.4, line 36 to col.5, line9, Nitta). It would have been obvious to one of ordinary skill in the art at the time of the invention was made to modify the combination system of Adya and McClaughry to include the claimed feature as taught by Nitta. The motivation of doing so would have been to obtain the lock and execute the processing in the lump can offer higher degree of freedom for processing and easily suppress the overhead (col.3, lines 35-38, Nitta).

Regarding claim 14, Adya disclose a file system, comprising:

- a memory ("206", Fig.2, Adya) configured to store information regarding directories and files organized as nodes in a namespace tree (elements "206"and "260" and ¶ [0056]-[0059], Adya); and
- a processor connected to the memory (Fig.2 and ¶ [0056]-[0059], Adya) and configured to:
    - o associate a read-write lock with each of the nodes in the namespace tree, acquire one or more first locks on names of one or more of the directories involved in a first operation (¶ [0033]; [0035]; [0037] and col.12, table II, Adya),
    - o determine whether the one or more first locks or the second lock conflict with one or more third locks acquired by a second operation (¶[0121]-[0123], Adya), and
    - o permit the first operation to execute when the one or more first locks or the second lock do not conflict with the one or more third locks, the one or more first

locks, the second lock, and the one or more third locks being read-write locks (¶[0123]; [0131]-[0133] and [0137], Adya).

Adya, however, does not disclose acquiring a lock on an entire pathname. McClaughry, on the other hand, discloses arbitrating concurrent access to file system objects including locks in the file system object hierarchy (abstract; summary and col.5, line 59 to col.6, line 62, McClaughry) and thus reads on the claimed "lock on an entire pathname". Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention was made to utilize the lock on the entire pathname into the distributed file system of Adya to derive the invention as claimed. The motivation of doing so would have been desirable to provide techniques and systems for managing file system object in a multithreaded environment in which multiple threads can simultaneously operate on the same object whenever possible to maximize throughput and availability of objects to a user but without causing object incoherency and inconsistent file operation results (col.2, lines 50-56, McClaughry).

Adya and McClaughry combination does not disclose serializing performance of the first and second operations when the first lock or the second lock conflicts with the third locks. Nitta discloses method and apparatus for releasing and obtaining shared and exclusive locks including serializing performance of the processes (col.3, lines 41-57 and col.4, line 36 to col.5, line9, Nitta). It would have been obvious to one of ordinary skill in the art at the time of the invention was made to modify the combination system of Adya and McClaughry to include the claimed feature as taught by Nitta. The motivation of doing so would have been

to obtain the lock and execute the processing in the lump can offer higher degree of freedom for processing and easily suppress the overhead (col.3, lines 35-38, Nitta).

Regarding claim 15, Adya discloses a method for performing first and second operations within a file system, comprising:

- acquiring one or more first locks on one or more first directory names involved in the first operation (¶[0117]-[0119], Adya),
- acquiring one or more second locks on one or more second directory names involved in the second operation (¶[0117]-[0119], Adya);
- determining whether the first and third locks conflict with the second and fourth locks (¶[0121]-[0123], Adya); and
- concurrently performing the first and second operations when the first and third locks do not conflict with the second and fourth locks, the one or more first locks, the one or more second locks, the third lock, and the fourth lock being read-mite locks (¶[0123]; [0131]-[0133] and [0137], Adya).

Adya, however, does not disclose acquiring a third or fourth lock on an entire pathname. McClaughry, on the other hand, discloses arbitrating concurrent access to file system objects including locks in the file system object hierarchy (abstract; summary and col.5, line 59 to col.6, line 62, McClaughry) and thus reads on the claimed "lock on an entire pathname". Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention was made to utilize the lock on the entire pathname into the distributed file system of Adya to derive the invention as claimed. The motivation of doing so would have been desirable to provide techniques and systems for managing file system object in a

multithreaded environment in which multiple threads can simultaneously operate on the same object whenever possible to maximize throughput and availability of objects to a user but without causing object incoherency and inconsistent file operation results (col.2, lines 50-56, McClaughry).

Adya and McClaughry combination does not disclose serializing performance of the first and second operations when the first lock or the second lock conflicts with the third locks. Nitta discloses method and apparatus for releasing and obtaining shared and exclusive locks including serializing performance of the processes (col.3, lines 41-57 and col.4, line 36 to col.5, line9, Nitta). It would have been obvious to one of ordinary skill in the art at the time of the invention was made to modify the combination system of Adya and McClaughry to include the claimed feature as taught by Nitta. The motivation of doing so would have been to obtain the lock and execute the processing in the lump can offer higher degree of freedom for processing and easily suppress the overhead (col.3, lines 35-38, Nitta).

Regarding claim l 7, Adya discloses a file system, comprising:

- a memory ("206", Fig.2, Adya) configured to store information regarding a plurality of directories and files as nodes in a namespace tree (elements "206"and "260" and ¶ [0056]-[0059], Adya); and
- a processor connected to the memory (Fig.2 and ¶ [0056]-[0059], Adya) and configured to:
    - o associate a read-write lock with each of the nodes in the namespace tree (¶ [0033]; [0035]; [0037] and col.12, table II, Adya),

- identify a set of the nodes involved in an operation, the identified nodes forming a pathname associated with the operation (¶ [0081]; [0033]-[0037]; [0104]-[0106] and [0118]-[0119], Adya),
- acquire a first one or more read-write locks, as one or more first locks, on the identified nodes (¶ [0033]; [0035]; [0037], Adya),
- determine whether the one or more first locks or the second lock conflict with a read-write lock acquired by another operation (¶[0121]-[0123], Adya), and
- permit the operation to execute when the one or more first locks and the second lock do not conflict with the other read-write locks (¶[0123]; [0131]-[0133] and [0137], Adya).

Adya, however, does not disclose acquiring locks on an entire pathname. McClaughry, on the other hand, discloses arbitrating concurrent access to file system objects including locks in the file system object hierarchy (abstract; summary and col.5, line 59 to col.6, line 62, McClaughry) and thus reads on the claimed "lock on an entire pathname". Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention was made to utilize the lock on the entire pathname into the distributed file system of Adya to derive the invention as claimed. The motivation of doing so would have been desirable to provide techniques and systems for managing file system object in a multithreaded environment in which multiple threads can simultaneously operate on the same object whenever possible to maximize throughput and availability of objects to a user but without causing object incoherency and inconsistent file operation results (col.2, lines 50-56, McClaughry).

Adya and McClaughry combination does not disclose serially execute the operation and the other operation when at least one of the one or more first lock or the second lock conflicts with the read-write lock acquired by the other operation. Nitta discloses method and apparatus for releasing and obtaining shared and exclusive locks including serializing performance of the processes (col.3, lines 41-57 and col.4, line 36 to col.5, line9, Nitta). It would have been obvious to one of ordinary skill in the art at the time of the invention was made to the combination system of Adya and McClaughry to include the claimed feature as taught by Nitta. The motivation of doing so would have been to obtain the lock and execute the processing in the lump can offer higher degree of freedom for processing and easily suppress the overhead (col.3, lines 35-38, Nitta).

Conclusion

5. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

6. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Hanh B. Thai whose telephone number is 571-272-4029. The examiner can normally be reached on 8 AM - 4:30 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Don Wong can be reached on 571-272-1834. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Hanh B Thai
Examiner
Art Unit 2163

July 11, 2006

DON WONG
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100